UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/619,644	07/14/2003	David Fritz	ZIL-553	5992

47713 7590 05/17/2007

IMPERIUM PATENT WORKS
P.O. BOX 587
SUNOL, CA 94586

EXAMINER	
LOHN, JOSHUA A	

ART UNIT	PAPER NUMBER
2114	

MAIL DATE	DELIVERY MODE
05/17/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

PTOL-90A (Rev. 04/07)

Office Action Summary	Application No.	Applicant(s)	
	10/619,644	FRITZ ET AL.	
	Examiner	Art Unit	
	Joshua A. Lohn	2114	

-- *The MAILING DATE of this communication appears on the cover sheet with the correspondence address* --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *28 February 2007*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-36* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-21 and 23-34* is/are rejected.

7)☒ Claim(s) *22,35 and 36* is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *14 July 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Response to Arguments

Applicant's arguments with respect to claims 1-30 have been considered but are moot in view of the new ground(s) of rejection.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Claims 11, 15, 20, 21, 25, 26, and 32-34 are rejected under 35 U.S.C. 102(e) as being anticipated by Rajsuman et al., United States Patent number 6,651,204, filed June 1, 2000.

As per claim 11, Rajsuman discloses *a method, comprising: receiving a script from a host computer onto a hardware debugging device* (Rajsuman, col. 6, lines 27-36, where the script is the event files), *the script defining a debugging action, the debugging action requiring a plurality of sub-actions be performed* (Rajsuman, col. 5, lines 15-22, where the event files are used to execute a multi-step test algorithm)*; interpreting the script; generating a plurality of microcommands from the script; and sending the plurality of microcommands to a target device, the microcommands causing the target device to perform the plurality of sub-actions* (Rajsuman,

col. 6, lines 27-35, where the event files are interpreted to generate microcommands, in the form of physical waveforms, where each aspect of the waveform would be a target sub-action).

As per claim 15, Rajsuman further discloses *the method of Claim 11, wherein said interpreting of the script occurs on the hardware debugging device, and wherein the script includes a statement that causes data to be sent from the hardware debugging device to the host computer* (Rajsuman, col. 6, lines 51-59).

As per claim 20, Rajsuman discloses *a debugging device comprising:*
a first communication interface that couples the debugging device to a host computer (Rajsuman, figure 2, where the debugging device is the pin unit, 25, the host is the workstation and library, 35 and 37, and the first communication interface is the connection between 37 and 24, also see column 5, lines 4-15, and col. 6, lines 25-32)*; a second communication interface that couples the debugging device to a target device* (Rajsuman, figure 2, where the second communication interface is the connection from 25 to 28)*; and means for receiving a script from the host computer, the script defining a debugging action to be taken with respect to the target device* (Rajsuman, col. 6, lines 27-29, where the script is the event test data)*, the debugging action requiring a plurality of sub-actions to occur* (Rajsuman, col. 6, lines 29-36, where each waveform transition is a sub-action)*, the means also being for interpreting the script and generating therefrom a plurality of microcommands that are sent to the target device* (Rajsuman, col. 6, lines 29-36, where the applied test waveforms come as a result of interpreting the script and generating the microcommands, in the form of waveform transitions).

As per claim 21, Rajsuman further discloses *the debugging device of Claim 20, wherein the microcommands are performed by the target device such that the sub-actions occur* (Rajsuman, col. 6, lines 32-34).

As per claim 25, Rajsuman discloses a *debugging device comprising: a first communication interface that couples the debugging device to a host computer* (Rajsuman, figure 2, where the debugging device is the pin unit, 25, the host is the workstation and library, 35 and 37, and the first communication interface is the connection between 37 and 24, also see column 5, lines 4-15, and col. 6, lines 25-32); *a second communication interface that couples the debugging device to a target device* (Rajsuman, figure 2, where the second communication interface is the connection from 25 to 28); *and a script interpreter executing on the debugging device, the script interpreter receiving a script from the host computer via the first communication interface, the script interpreter interpreting the script and causing the debugging device to communicate with the target device over the second communication interface such that a debugging action is performed* (Rajsuman, col. 6, lines 25-36, where the pin unit is the script interpreter, which interprets the event data that acts as a script, and communicates the interpreted data to the target in the form of waveforms to execute the debugging action).

As per claim 26, Rajsuman further discloses *the debugging device of Claim 25, wherein there is no operating system stored on the debugging device* (Rajsuman, col. 6, lines 25-36, where the debugging device is merely a conversion apparatus, and no operating system is disclosed).

As per claim 32, Rajsuman further discloses *the method of Claim 11, wherein the script comprises ASCII text characters* (Rajsuman, col. 5, lines 43-51, where the event format consists of ASCII text characters, this is inherent in the definition the event format where numerical and alphabetical characters are used to define events in the event format, see copending application 10/089,137, now patent number 6,978,410, for definition).

As per claim 33, Rajsuman further discloses *the debugging device of Claim 20, wherein the script comprises ASCII text characters* (Rajsuman, col. 5, lines 43-51, where the event format consists of ASCII text characters).

As per claim 34, Rajsuman further discloses *the debugging device of Claim 25, wherein the script comprises ASCII text characters* (Rajsuman, col. 5, lines 43-51).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 1-3, 5-8, 12, 16, 18, 19, 23, 24, 28, 29, and 31 are rejected under 35 U.S.C. 103(a) as being unpatentable over Rajsuman et al., United States Patent number 6,651,204, filed June 1, 2000, in view of Tegethoff, United States Patent number 5,937,154, published August 10, 1999.

As per claim 1, Rajsuman discloses a *debugging device comprising: a first communication interface that couples the debugging device to a host computer* (Rajsuman, figure 2, where the debugging device is the pin unit, 25, the host is the workstation and library, 35 and 37, and the first communication interface is the connection between 37 and 24, also see column 5, lines 4-15, and col. 6, lines 25-32); *a second communication interface that couples the debugging device to a target device* (Rajsuman, figure 2, where the second communication interface is the connection from 25 to 28); *and a script interpreter executing on the debugging device, the script interpreter receiving a script from the host computer via the first communication interface* (Rajsuman, col. 6, lines 25-27, where the script is disclosed in the event files)*, the script interpreter interpreting the script and causing the debugging device to communicate with the target device over the second communication interface* (Rajsuman, col. 6, lines 29-32, where the events are converted and executed). Rajsuman fails to disclose that the script defines a loop that involves a plurality of reads that are then executed.

Tegethoff discloses *a script defining a loop* (Tegethoff, col. 11, lines 1-25) *that involves performing a plurality of reads* (Tegethoff, col. 11, line 42 through col. 12, line 3).

It would have been obvious to one of ordinary skill in the art at the time of the invention to use the looping reads of Tegethoff as some of the events of Rajsuman.

This would have been obvious because Rajsuman discloses a desire to execute algorithmic pattern generator to test memory devices (Rajsuman, col. 5, lines 1-3), and Tegethoff discloses such an algorithmic pattern generator in the walking ones address test (Tegethoff, col. 10, lines 61-67). Therefor it would have been obvious to use the specified test algorithm of

Tegethoff in the event system of Rajsuman to provide a specific test algorithm to execute the type of testing desired.

As per claim 2, Rajsuman and Tegethoff further disclose *the debugging device of Claim 1, wherein the script includes a loop statement, and wherein the loop statement includes an expression and at least one statement* (Tegethoff, col. 11, lines 1-23).

As per claim 3, Rajsuman and Tegethoff further disclose the *debugging device of Claim 2, wherein said at least one statement is a read statement* (Tegethoff, col. 11, line 42 through col. 12, line 17).

As per claim 5, Rajsuman and Tegethoff further disclose *the debugging device of Claim 1, wherein the performing of the plurality of reads results in an amount of data being retrieved from the target device, and wherein the script includes a statement that causes the amount of data to be sent from the debugging device to the host computer* (Rajsuman, col. 6, lines 51-59).

As per claim 6, Rajsuman and Tegethoff further disclose *the debugging device of Claim 2, wherein the script includes a second statement in addition to said at least one statement* (Tegethoff, col. 11, lines 1-23).

As per claim 7, Rajsuman and Tegethoff further disclose *the debugging device of Claim 1, wherein the script is not compiled on the host computer, and wherein the script is not compiled on the debugging device* (Rajsuman, col. 7, lines 33-38, where the events are compiled and generated off-line, independent of the host computer and debugging device).

As per claim 8, Rajsuman and Tegethoff further disclose *the debugging device of Claim 1, wherein there is no operating system stored on the debugging device* (Rajsuman, col. 6, lines 25-36, where the debugging device is merely a conversion apparatus, and no operating system is disclosed).

As per claim 12, Rajsuman discloses the method of claim 11, but fails to disclose that the debugging action includes a read of a block of memory locations on the target, wherein the sub-actions are a read of one of the memory locations.

Tegethoff discloses *wherein the debugging action includes a read of a block of memory locations on the target device* (Tegethoff, col. 11, line 10)*, and wherein one of the plurality of sub-actions is a read of one of the memory locations* (Tegethoff, col. 11, line 35 through col. 12, line 2).

It would have been obvious to one of ordinary skill in the art at the time of the invention to use the looping reads of Tegethoff as some of the events of Rajsuman.

This would have been obvious because Rajsuman discloses a desire to execute algorithmic pattern generator to test memory devices (Rajsuman, col. 5, lines 1-3), and Tegethoff discloses such an algorithmic pattern generator in the walking ones address test (Tegethoff, col. 10, lines 61-67). Therefor it would have been obvious to use the specified test algorithm of Tegethoff in the event system of Rajsuman to provide a specific test algorithm to execute the type of testing desired.

As per claim 16, Rajsuman discloses the method of claim 11, but fails to disclose that the script includes a loop statement, an arithmetic operator, and a variable.

Tegethoff discloses a loop statement (Tegethoff, col. 11, line 7), an arithmetic operator (Tegethoff, col. 11, line 51, where "=" is an arithmetic operator), and a variable (Tegethoff, col. 11, lines 14-415, where the difference result is a variable).

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the statements of Tegethoff in the invention of Rajsuman for the reasons mentioned above with respect to claim 12.

As per claim 18, Rajsuman discloses the method of claim 11, but fails to disclose that the script includes a break statement.

Tegethoff discloses *a break statement* (Tegethoff, col. 11, line 23, where the "done" is a break statement).

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the statements of Tegethoff in the invention of Rajsuman for the reasons mentioned above with respect to claim 12.

As per claim 19, Rajsuman discloses the method of claim 11, but fails to disclose that the script includes a boolean operator.

Tegethoff discloses *a boolean operator* (Tegethoff, col. 11, line 15, where the "if" comparison involves a Boolean operator).

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the statements of Tegethoff in the invention of Rajsuman for the reasons mentioned above with respect to claim 12.

As per claim 23, Rajsuman discloses the method of claim 20, but fails to disclose that one of the sub-actions involves setting a breakpoint.

Tegethoff discloses *setting a breakpoint* (Tegethoff, col. 11, line 23, where the "done" is a breakpoint between tests, col. 14, lines 34-38).

It would have been obvious to one of ordinary skill in the art at the time of the invention to use the looping reads of Tegethoff as some of the events of Rajsuman.

This would have been obvious because Rajsuman discloses a desire to execute algorithmic pattern generator to test memory devices (Rajsuman, col. 5, lines 1-3), and Tegethoff discloses such an algorithmic pattern generator in the walking ones address test (Tegethoff, col. 10, lines 61-67). Therefor it would have been obvious to use the specified test algorithm of Tegethoff in the event system of Rajsuman to provide a specific test algorithm to execute the type of testing desired.

As per claim 24, Rajsuman discloses the method of claim 20 and further discloses a designation of a register internal to a processor of the target device in the script (Rajsuman, col. 5, lines 64-65, where the memory specified to the test can be embedded with a processor, which includes register memories), but fails to disclose that the script includes a loop statement and an arithmetic operator.

Tegethoff discloses a loop statement (Tegethoff, col. 11, line 7) and an arithmetic operator (Tegethoff, col. 11, line 51, where "=" is an arithmetic operator).

It would have been obvious to one of ordinary skill in the art at the time of the invention to combine the statements of Tegethoff in the invention of Rajsuman for the reasons mentioned above with respect to claim 23.

As per claim 28, Rajsuman discloses the debugging device of claim 25, but fails to disclose the inclusion of a loop statement.

Tegethoff disclose *the debugging device of Claim 1, wherein the script includes a loop statement, and wherein the loop statement includes an expression and at least one statement* (Tegethoff, col. 11, lines 1-23).

It would have been obvious to one of ordinary skill in the art at the time of the invention to use the looping reads of Tegethoff as some of the events of Rajsuman.

This would have been obvious because Rajsuman discloses a desire to execute algorithmic pattern generator to test memory devices (Rajsuman, col. 5, lines 1-3), and Tegethoff discloses such an algorithmic pattern generator in the walking ones address test (Tegethoff, col. 10, lines 61-67). Therefor it would have been obvious to use the specified test algorithm of Tegethoff in the event system of Rajsuman to provide a specific test algorithm to execute the type of testing desired.

As per claim 29, Rajsuman and Tegethoff further disclose the *debugging device of Claim 2, wherein said at least one statement is a read statement* (Tegethoff, col. 11, line 42 through col. 12, line 17).

As per claim 31, Rajsuman and Tegethoff further disclose *the debugging device of Claim 1, wherein the script comprises ASCII text characters* (Rajsuman, col. 5, lines 43-51, where the event format consists of ASCII text characters)

Claims 4 and 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Rajsuman in view of Tegethoff in view of "The Zen of Diagnostics", Published in Embedded Systems Programming, June 1990.

As per claim 4, Rajsuman and Tegethoff disclose the debugging device of claim 3, but fail to explicitly discloses that the expression in the statement includes a number specifying a number of times to perform a read statement.

"The Zen of Diagnostics" discloses a system that specifies the number of times to perform a read statement ("The Zen of Diagnostics", page7, where the length expression indicates the number of times to perform a read statement when checking the memory).

It would have been obvious to one skilled in the art at the time of the invention to use the read counter of "The Zen of Diagnostics" in the invention of Rajsuman and Tegethoff.

This would have been obvious because the expression of a number of read statements to perform is included in a ram test code program ("The Zen of Diagnostics", figure 1). Further, "The Zen of Diagnostics" discloses a need to examine a system analytically in an effort to look for all possible failure modes ("The Zen of Diagnostics, page 6), as is presented in part by the

RAM test code of figure 1. It would have obviously benefited Rajsuman and Tegethoff in the desire to present complete physical and functional fault detection (Rajsuman, col. 6, lines 12-17).

As per claim 30, Rajsuman and Tegethoff disclose the debugging device of claim 29, but fail to explicitly discloses that the expression in the statement includes a number specifying a number of times to perform a read statement.

"The Zen of Diagnostics" discloses a system that specifies the number of times to perform a read statement ("The Zen of Diagnostics", page7, where the length expression indicates the number of times to perform a read statement when checking the memory).

It would have been obvious to one skilled in the art at the time of the invention to use the read counter of "The Zen of Diagnostics" in the invention of Rajsuman and Tegethoff.

This would have been obvious because the expression of a number of read statements to perform is included in a ram test code program ("The Zen of Diagnostics", figure 1). Further, "The Zen of Diagnostics" discloses a need to examine a system analytically in an effort to look for all possible failure modes ("The Zen of Diagnostics, page 6), as is presented in part by the RAM test code of figure 1. It would have obviously benefited Rajsuman and Tegethoff in the desire to present complete physical and functional fault detection (Rajsuman, col. 6, lines 12-17).

Claims 9, 10, 13, and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Rajsuman in view of Tegethoff in view of Cromer et al., United States Patent number 6,263,373, published July 17, 2001.

As per claim 9, Rajsuman and Tegethoff disclose the debugging device of claim 1, with a second communication interface to couple the debugging device to the target device, however they fail to disclose where the target device comprises an on-chip debugging circuit, and wherein the second interface couples the debugging device to the on-chip debugging circuit.

Cromer discloses a target device including an on-chip debugging circuit (Cromer, col. 3, lines 21-30, where the TAP provides an on-chip debugging circuit) and also discloses connecting an interface to this debugging circuit (Cromer, figure 2, where the on-chip debugging circuit, 201 is connected to a debugging circuit through the network).

It would have been obvious to one skilled in the art at the time of the invention to include the on-chip debugging circuit of Cromer in the invention of Rajsuman and Tegethoff.

This would have been obvious because Cromer discloses that the test access port debugging circuit is useful in code development and problem resolution to allow access to the processor utilizing a known architecture standard (Cromer, col. 1, lines 55-65). Rajsuman shows a desire to have thorough problem resolution (Rajsuman, col. 6, lines 13-17), and utilize standard connectors in the course of this testing (Rajsuman, col. 3, line 66 through col. 4, line 9). Therefor, the on-chip debugging circuit would provide the obvious benefit of giving Rajsuman a specific interface type that aids in the thorough testing desired.

As per claim 10, Rajsuman and Tegethoff disclose *the debugging device of Claim 1, wherein the script is communicated from the host computer to the debugging device* (Rajsuman, col. 6, lines 25-28). Rajsuman and Tegethoff fail to disclose transmitting the script as a network packet.

Cromer discloses sending a test program over a network packet (Cromer, col. 3, lines 31-38).

It would have been obvious to one skilled in the art at the time of the invention to implement the network method of Cromer in the invention of Rajsuman and Tegethoff.

This would have been obvious because Rajsuman discloses the desire to ability to run test programs from an external workstation (Rajsuman, figure 2) to a plurality of devices (Rajsuman, col. 2, lines 33-36). Cromer provides the ability to allow this workstation to be placed at any location that can be accessed with a network and still provide the ability to test a plurality of devices, which provides an additional benefit of reduced cost by providing debugging across a wider network (Cromer, col. 2, lines 5-18). Thus it would have been obvious to use the remote testing of Cromer in the invention of Rajsuman to provide the obvious benefit of allowing the testing to occur regardless of geographical separation.

As per claim 13, Rajsuman and Tegethoff disclose the method of claim 12, wherein the target device includes a processor (Rajsuman, col. 2, lines 43-47) in which the microcommands are executed, however they fail to disclose where the target device includes an on-chip debugging circuit executing these commands.

Cromer discloses a target device including an on-chip debugging circuit (Cromer, col. 3, lines 21-30, where the TAP provides an on-chip debugging circuit) and also discloses connecting an interface to this debugging circuit (Cromer, figure 2, where the on-chip debugging circuit, 201 is connected to a debugging circuit through the network).

It would have been obvious to one skilled in the art at the time of the invention to include the on-chip debugging circuit of Cromer in the invention of Rajsuman and Tegethoff.

This would have been obvious because Cromer discloses that the test access port debugging circuit is useful in code development and problem resolution to allow access to the processor utilizing a known architecture standard (Cromer, col. 1, lines 55-65). Rajsuman shows a desire to have thorough problem resolution (Rajsuman, col. 6, lines 13-17), and utilize standard connectors in the course of this testing (Rajsuman, col. 3, line 66 through col. 4, line 9). Therefor, the on-chip debugging circuit would provide the obvious benefit of giving Rajsuman a specific interface type that aids in the thorough testing desired.

As per claim 14, Rajsuman and Tegethoff disclose the method of claim 12 with the execution of microcommands, but fail to disclose the target device including a JTAG interface to receive these commands.

Cromer discloses a target device including a target device including a JTAG interface (Cromer, col. 3, lines 21-30, where the TAP provides an on-chip debugging circuit, and col. 1, lines 60-62, where it is shown that the TAP is JTAG compatible) and executing commands to this debugging circuit (Cromer, figure 2, where the on-chip debugging circuit, 201 is connected to a debugging circuit through the network).

It would have been obvious to one skilled in the art at the time of the invention to include the on-chip debugging circuit of Cromer in the invention of Rajsuman.

This would have been obvious because Cromer discloses that the test access port debugging circuit is useful in code development and problem resolution to allow access to the processor utilizing a known architecture standard (Cromer, col. 1, lines 55-65). Rajsuman shows a desire to have thorough problem resolution (Rajsuman, col. 6, lines 13-17), and utilize standard connectors in the course of this testing (Rajsuman, col. 3, line 66 through col. 4, line 9). Therefor, the on-chip debugging circuit would provide the obvious benefit of giving Rajsuman a specific interface type that aids in the thorough testing desired.

Claims 17 and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Rajsuman in view of Cromer et al., United States Patent number 6,263,373, published July 17, 2001.

As per claim 17, Rajsuman discloses the method of claim 11, but fails to disclose the inclusion of a sleep statement in the script.

Cromer discloses a sleep statement for the testing system (Cromer, col. 7, line 61 through col. 8, line 3, where the magic packet detection means that any packet destined for an operating system is ignored, resulting in a sleeping state for the debug system).

It would have been obvious to one skilled in the art at the time of the invention to include the sleep statement of Cromer in the invention of Rajsuman.

This would have been obvious because the ability of Cromer to provide communications to the system that are both debugging and normal operations (Cromer, col. 7, line 61 through col. 8, line 20) would allow for the improvement to the debugging of Rajsuman by allowing the system to be used in normal operation and not just in a design testing environment (Rajsuman, col. 1, lines 5-10).

As per claim 27, Rajsuman discloses the device of claim 25, but fails to disclose the use of a sleep statement in the script.

Cromer discloses a sleep statement for the testing system (Cromer, col. 7, line 61 through col. 8, line 3, where the magic packet detection means that any packet destined for an operating system is ignored, resulting in a sleeping state for the debug system).

It would have been obvious to one skilled in the art at the time of the invention to include the sleep statement of Cromer in the invention of Rajsuman.

This would have been obvious because the ability of Cromer to provide communications to the system that are both debugging and normal operations (Cromer, col. 7, line 61 through col. 8, line 20) would allow for the improvement to the debugging of Rajsuman by allowing the system to be used in normal operation and not just in a design testing environment (Rajsuman, col. 1, lines 5-10).

Allowable Subject Matter

Claims 22, 35, and 36 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure is provided on form PTO-892.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Joshua A. Lohn whose telephone number is (571) 272-3661. The examiner can normally be reached on M-F 8-4.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Scott Baderman can be reached on (571) 272-3644. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Joshua A Lohn
Patent Examiner
Art Unit 2114